IN THE SPECIFICATION

Please amend the paragraph at page 20, lines 14-22, as follows:

--Figure 10 shows a flowchart of a simple method to remove the unwanted small gradient values. The gradient values are tested as shown in 901. If the absolute value is less than *GRADTHRES*, it is set to zero as shown in 904. If the value is greater than *GRADTHRES* and positive, it will be set to a positive number. If the value is greater than *GRADTHRES* and negative, it will be set to a negative number. Here, +10 and –10 are used, respectively, as an example. This process is shown in 902 through [[905]] 906. In the end, the gradient threshold function will produce positive and negative pulses such as those shown in 1301 through 1305.--

Please amend the paragraph at page 23, lines 6-15, as follows:

--The pitch values detected may vary due to the failure of a user to maintain the pitch within a single note. Step 1115 checks whether the count (compare step 1102) is greater than 4. The FindDom function as shown in [[1116]] 1117 finds the dominant pitch value. In this invention, the detected pitch values are corrected to the nearest MIDI number in 1118. The MIDI number is computed as:

$$hum\_des[j].pitch = 49 + \frac{floor\left[12\log(\frac{\det ected\_pitch}{440})\right]}{\log 2}$$

The *floor(x)* function returns a floating-point value representing the largest integer that is less than or equal to *x*. The process is repeated until all notes in the input data have their pitch detected as shown in 1119.--

Please amend the paragraph at page 24, lines 15-27, as follows:

--The pitch values of the note under test are contained in the array *pitch*. The process 1212 compares the absolute distance of the pitch value from the two centres. The pitch value is added to the accumulators called, *temp1* or *temp2* depending on the result of the comparison as shown in 1213 and 1214. This process repeats until all the pitch values in the note are tested as shown in 1215. When the test in 1215 yields a "No," it is tested at 1216 and 1217 whether count 1 and count 2 (compare step 1211) > 0 respectively. The new centres are computed and the member counts are incremented as shown in 1218 and 1219. They are the average of the member pitch values. The processes 1220 and 1221 test if the two centres change. If the two centres do not change, the iteration stops immediately. If there are changes in any of the centres, the iteration of the processes from 1211 through 1221 repeat until the maximum number of loops (MAXLOOP) has been reached as tested in step 1222. The maximum number of loops is 10 in this exemplary embodiment.--

Please amend the paragraph from page 24, line 29, to page 25, lines 1-5, as follows:

--If the numbers of members of the two centres is close, as tested in 1223, the average of the two centres is returned as the dominant pitch. If they are not close enough and count 1 > count 2 as determined at step 1224, the centre with the larger number of members is returned as the dominant pitch as shown in 1225 through 1227. In this way, the cluster with the highest number of members is classified as the prominent cluster while the other cluster is classified as the outlier cluster. The pitch of the note is set to the centre of the prominent cluster.—